2D Mapping of Cluttered Indoor Environments by Means of 3D Perception

Oliver Wulf [1], Kai O. Arras [2], Henrik I. Christensen [2] and Bernardo Wagner [1]

[1] *Institute for Systems Engineering (ISE/RTS)*
University of Hannover
Hannover, Germany
{wulf,wagner}@rts.uni-hannover.de

[2] *Centre for Autonomous Systems (CAS)*
Royal Institute of Technology
Stockholm, Sweden
{kai-a,hic}@nada.kth.se

***Abstract*— This paper presents a combination of a 3D laser sensor and a line-base SLAM algorithm which together produce 2D line maps of highly cluttered indoor environments. The key of the described method is the replacement of commonly used 2D laser range sensors by 3D perception. A straightforward algorithm extracts a virtual 2D scan that also contains partially occluded walls. These virtual scans are used as input for SLAM using line segments as features. The paper presents the used algorithms and experimental results that were made in a former industrial bakery. The focus lies on scenes that are known to be problematic for pure 2D systems. The results demonstrate that mapping indoor environments can be made robust with respect to both, poor odometry and clutter.**

I. INTRODUCTION

In many typical robot settings for indoor navigation, a range sensor takes measurements in a plane parallel to the floor. The supposition that this cross-section contains enough information for the robot to navigate is commonly called the “2D assumption”. However, even structured indoor environments can be hard to model if dynamic objects and clutter dominate over the static structures which are good for navigation. More sophisticated feature extraction algorithms or the explicit modeling of dynamic objects during the mapping process [4] are examples of worthwhile approaches to be explored. But when the amount of clutter increases (Fig. 1), the cross-section contains more and more insufficient and misleading information for localization and mapping. The 2D assumption cannot be held anymore. Another approach is the use of 3D laser range data. In combination with mobile robots for indoor, two types of 3D sensors are already in use. One group is using a fixed mounted 2D scanner. In this case the robots are moved for 3D scanning [1]. These kind of 3D scanners are used to build static models of the environment. They cannot be used for navigation, as they already require an accurate 2D localization to accumulate the 3D data. The second group of 3D sensors has got an extra servo drive to turn a 2D scanner independently of the robot platform. This kind of sensor can be used for navigation like it is described in this paper. Nevertheless current applications [2] [3] are only used for static modeling up to now. This can be attributed to relatively long scanning times and the need to stand still during the scanning process.

The introduction of 3D sensors into navigation was first done for mobile outdoor robots, as the disadvantages of 2D sensors are especially distinct in these environments. One procedure in this application is to build up full 3D navigation systems including a 3D sensor and a 3D or 2½D map representations [4][5]. These systems can be used in completely unstructured environments but they have the disadvantage of being computationally very expensive. An alternative method for 3D outdoor navigation is described in [6]. The system combines a 3D sensor with a 2D SLAM algorithm and map representation for semi structured outdoor environments. By this means it takes advantage of 3D perception in combination with less complex 2D navigation algorithms.

Figure 1. Cluttered indoor scene (CAS living room)

This paper pursues a similar strategy as it is on 2D navigation with 2D range data, but it releases the 2D assumption by processing 3D sensory data and projecting them into a virtual plane. Avoiding the complexity of full 3D models, we build 2D maps of cluttered indoor environments with a combination of a new 3D sensor and a line-based SLAM algorithm. Section II describes a 3D sensor that can be used for mobile robot navigation and virtual 2D scans that are used to interface between the 3D and the 2D world. An exemplary feature-based SLAM algorithm that is used to process 2D line maps is described in section III. The practical results of a series of experiments at an industrial building are described and illustrated in section IV.

II. 3D Perception

The 3D perception system that is described in this paper consists of two parts. The first is a fast 3D laser range sensor that is especially adapted for use on mobile robots. This scanner is described in section IIa. The second part is a construction called virtual 2D scans. Section IIb will introduce these virtual scans and will also present an algorithm that is able to extract them out of 3D point-clouds.

A. Fast 3D Laser Range Scans

As there is no commercial 3D laser range scanner available which could be used for mobile robots, it is common practice to assemble 3D sensors out of a standard 2D scanner and an additional servo drive [7]. A scanner that is used in our system is the SICK LMS 291 in combination with a self-build servo drive (*scan drive*). Different orientations of the 2D scanner in combination with different turning axis result in a number of possible scanning patterns. Two scanning patterns that are practicable for our application are the *yawing scan*, vertical 2D raw scan and rotation around the upright axis (see Fig. 2a), and the *yawing scan top*, 2D raw scan facing up and rotation around the upright axis (see Fig. 2b). The yawing scan pattern results in the maximal possible field of view of 360° horizontal and 180° vertical. The yawing scan top has got also a horizontal opening angle of 360° but it covers only the upper hemisphere. For this reason a sensor with such a scanning pattern is not able to detect obstacles that lie on the ground. On the other hand the data is sufficient for localization and mapping and the scan time, which is half of the yawing scan time, is attractive for faster motion.

(a) (b)

Figure 2. Continuously turning 3D scanner:
(a) yawing scan, (b) yawing scan top

Short scanning times and the ability to scan while moving are the main criteria that decide on the usability of 3D scanners for navigation tasks. For this reason the sensor that is used in this paper contains a number of improvements. One mechanical improvement is the ability of the *scan drive* to turn continuously, which is implemented by using slip rings for power and data connection to the 2D scanner. This leads to a homogeneous distribution of scan points and saves the energy and time that is needed for acceleration and deceleration of panning scanners. Another improvement that becomes important with short scanning times of a few seconds is the compensation of systematic measurement errors. This compensation is done by means of sensor analysis and hard real-time synchronization and time stamping. The result of these optimizations that are described in detail in [7] lead to scan times as short as 4.8s for a yawing scan with 1° horizontal and 1° vertical resolution (see Fig. 3), 2.4s for a 2°, 1° yawing scan or a 1°, 1° yawing scan top or only 1.2s for a yawing scan top with 2°, 1° resolution. Another feature is the ability to scan while driving, which is achieved with a move compensation algorithm [6]. This algorithm is using a 3D dead-reckoning mechanism that combines wheel odometry and a gyroscope. The estimated robot position is used to transform the 3D point-cloud into a world fixed and therefore undistorted coordinate frame.

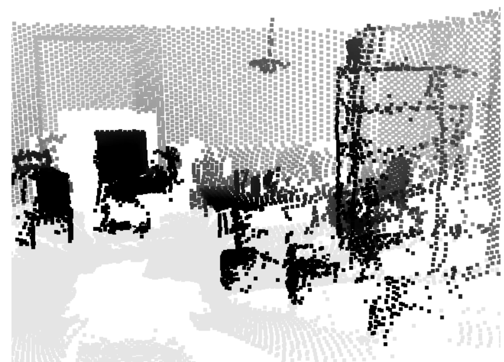

Figure 3. 3D point-cloud yawing scan 1°, 1°

B. Virtual 2D Scans

The 3D point-clouds that are acquired by the 3D scanner contain detailed information about the surrounding environment. Because 3D point-clouds are raw data representations, they include redundant information and many measurement points which are not needed for localization and mapping. Approaches which use this raw data for scan matching and full 3D modeling are computational very expensive. If the goal is to localize or navigate a mobile robot, these full 3D algorithms are not efficient. The use of virtual 2D scans is more efficient here as it aims to reduce the amount of data without loosing information that is essential for mobile robot localization. The reduced data sets can afterwards be used for computationally less expensive matching and SLAM algorithms. The data representation that is chosen of virtual 2D scans is similar to the data that can be measured directly with a 2D laser range sensor. It is defined as a number of object surface points that are given in a 2D robot coordinate system. For this reason existing 2D scanners can be replaced by more intelligent 3D perception systems and can be used by keeping existing 2D SLAM algorithms.

These intelligent sensors are based on algorithms that are able to extract the information that is essential for SLAM out of 3D point-clouds. This paper describes a first, straightforward, heuristic that extracts virtual scans from cluttered indoor scenes. Objects that are preferably used for indoor localization are walls because they are immobile and efficiently modeled as lines. The first step to create this virtual 2D scan is to project all 3D points onto the plane by setting the z-coordinate to zero. A virtual 2D scan that contains primarily walls can thereafter be assembled by taking one point out of each vertical raw scan (resp. two points for a yawing scan top). This point is chosen to be the one with the largest distant to the center of the robot. As walls build the boundary of a closed indoor scene the chosen point is most probably a wall point. By this means points lying on the floor, ceiling or on obstacles are filtered out. The thus generated 2D scan is only disturbed by open doors, windows and objects that cover the wall completely. Fig. 4 shows a virtual 2D scan of the CAS living room in comparison to a regular 2D range scan taken at 50cm height. A photograph and one view of the 3D point-cloud from the same room can be seen in Fig. 1 and Fig. 3.

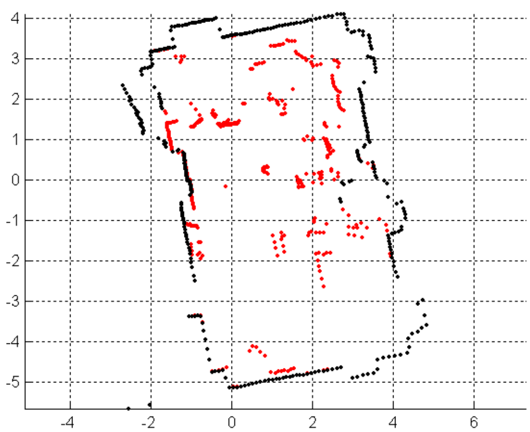

Figure 4. Virtual 2D scan (black), Regular 2D scan (50cm height, red)

III. LINE-BASED SLAM

This section describes the line-based SLAM algorithm which has been employed for the experiments. Assuming that the reader is familiar with the standard SLAM framework as initially proposed by [8], [9] we will focus on the particularity of our method.

A. Feature Representation

When lines are used as landmarks for SLAM the representation problem of uncertain features becomes visible. Several line models have shortcomings when used for SLAM such as the Hessian (alpha, r)-representation whose error propagation across frame transforms leads to physically meaningless covariance matrices. Here we use the SPmodel [10] to represent line segments. In the SPmodel, a reference frame is attached by default to each feature regardless its type. Errors are represented locally in that frame and in case of a non-full pose feature (e.g. lines or (x,y)-points) the so called binding matrix selects those rows in the feature state vector that "make sense" for the given feature type.

To represent finite line segments, the segment endpoints are expressed relative to the frame of the supporting line as single rank (i.e. one dimensional) locations with a constant uncorrelated typical uncertainty.

B. Robot Displacement

A differential drive configuration is used as the kinematic model for the robot. This simplification allows us to apply an uncertainty model that models non-systematic odometry errors (correctly) in the wheel space and not in the Cartesian space [11]. From the odometry data delivered by the wheel encoders for the robot x- and y-position and the IMU device for the vehicle orientation, the encoder values for the left and right wheel are determined and taken as inputs for the kinematic model. Given a higher frequency odometry update with respect to the rate of the observations, a recursive formulation yields the sought process model Jacobian. This Jacobian is needed to update the first row and column of the map covariance matrix during the state prediction step.

C. Observation

The line extraction algorithm from [12] has been used. It uses a sliding window technique and fit expressions that minimizes weighted squared errors perpendicular from the points onto the line. Its complexity is $O(n{*}n_w)$ where n is the number of scan points, n_w the size of the window and $n_w{<<}n$ holds. Collinear segments are fused in a final steps using an NNSF-like clustering algorithm based on a Mahalanobis distance matrix.

D. Data association

Line segments are an example of features with a physical extension. This plus of information can be used to support data association. After measurement prediction, based on the infinite line parameters, candidates are found which are compatible on a significance level alpha by means of the usual validation gate approach. Then, the candidate that has an overlap length greater than zero maximizing the overlap between the predicted map feature and the observation is chosen. For the calculation of the overlap length, endpoints are transformed into the frame of the map line and projected prependicularly.

Finally, integration uses an iterative EKF under a strongest observation policy. We first integrate those observations in the local map that are most certain thus fostering good filter convergence. This strategy helps to disambiguate remaining data association uncertainties during the integration process.

IV. EXPERIMENTS

This section shows the experimental results of two test runs made in a 100x100m industrial like indoor environment. The description includes the used robot system (section IVa), the test environment (section IVb) and the experimental results (section IVc).

A. Experimental Setup

The experimental robot consists of an iRobot ATRV based mobile platform for urban exploration and a 3D laser range scanner with external processing unit (see Fig. 5). The platform allows remote controlled driving via WLan. Further more the onboard unit calculates an odometric position estimation combining wheel encoders and a fiber optic gyro. The 3D scanner is built out of a Sick LMS 291 and a *scanDrive* which is a servo drive that is especially constructed for fast and continuous scanning. A scalable processing box (SPB [13]) that is based on an embedded-PC with Linux/RTAI real-time operating system is doing the 3D scanning and data acquisition. The ATRV onboard unit and the SPB are interconnected via CAN-Bus.

The mapping experiments were carried out with a remote controlled robot driving at about 0.1m/s. The 3D scanner is set up to measure full 180°x360° range scans with a resolution of 1°x1°. That results in a scan time of 4.8s. With the robot driving at the given speed the 3D scanner is able to measure one 3D scan, respectively one virtual 2D scan, about every 0.5m.

Within these first experiments, all 3D data processing and SLAM is done offline in a MATLAB environment.

Figure 5. Experiments in a former industrial bakery

B. *Test Environment*

A large indoor environment that was used as an industrial bakery before has been available for the experiments described in this paper is. Currently it is used as a training site by Swedish military, police and fire brigades.

The experiments covered two test areas of 20x30m and 30x30m with a number is interconnected rooms. The rooms have got a flat floor, but there are several steps between rooms. Though they are traversable with the ATRV, they lead to large odometry errors. A lot of pipes and electrical installations are mounted on the walls and on the ceiling (see Fig. 5). Various obstacles and 6 people were in the test area during the experiment. Especially because of the occluded walls the test area is known to be problematic for pure 2D mapping.

C. *Results*

The algorithm that is used to process virtual 2D scans was applied to all 208 3D point-clouds out of both test runs. The experiments show very good and stabile results for wall extraction in closed indoor scenes. This can be traced back to the fact that only one correct wall point per vertical 2D scan is needed to produce a correct representation in the virtual 2D scan. This removes a vast of obstacles. Only obstacles that fully occlude the wall, e.g. people walking closely to the robot cannot be removed. As the virtual 2D scan contains mostly wall points it turns out to be ideal input data for the following 2D algorithms. The output is comparable with a 2D scan taken in an empty room or corridor.

Fig. 6 shows a clipping of a typical scene in a corridor. In this case two persons, a ladder, open cupboard doors and several other obstacles occlude the corridor. As it can be seen in the lower part of Fig. 6, a normal 2D scan (red points) contains many measurement points on these obstacles in contrast to the virtual 2D scan (black points) that represents large parts of the walls.

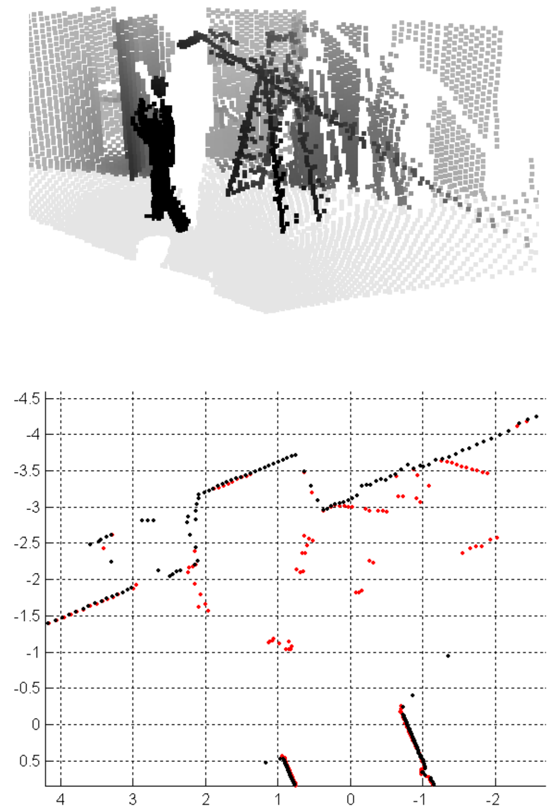

Figure 6. Experimental data of cluttered corridor

The algorithm was applied to build two maps. The first run, revisiting the start point twice, consists of 156 steps and results in a map with 103 landmarks (Fig. 7a & 7b). The second run, starting in a big hall, has 72 steps and yields a map with 126 features (Fig. 8a & 8b).

In addition to the clutter in the test environment, two circumstances made localization and mapping more difficult. On reason is the large odometry drift of the skid steering

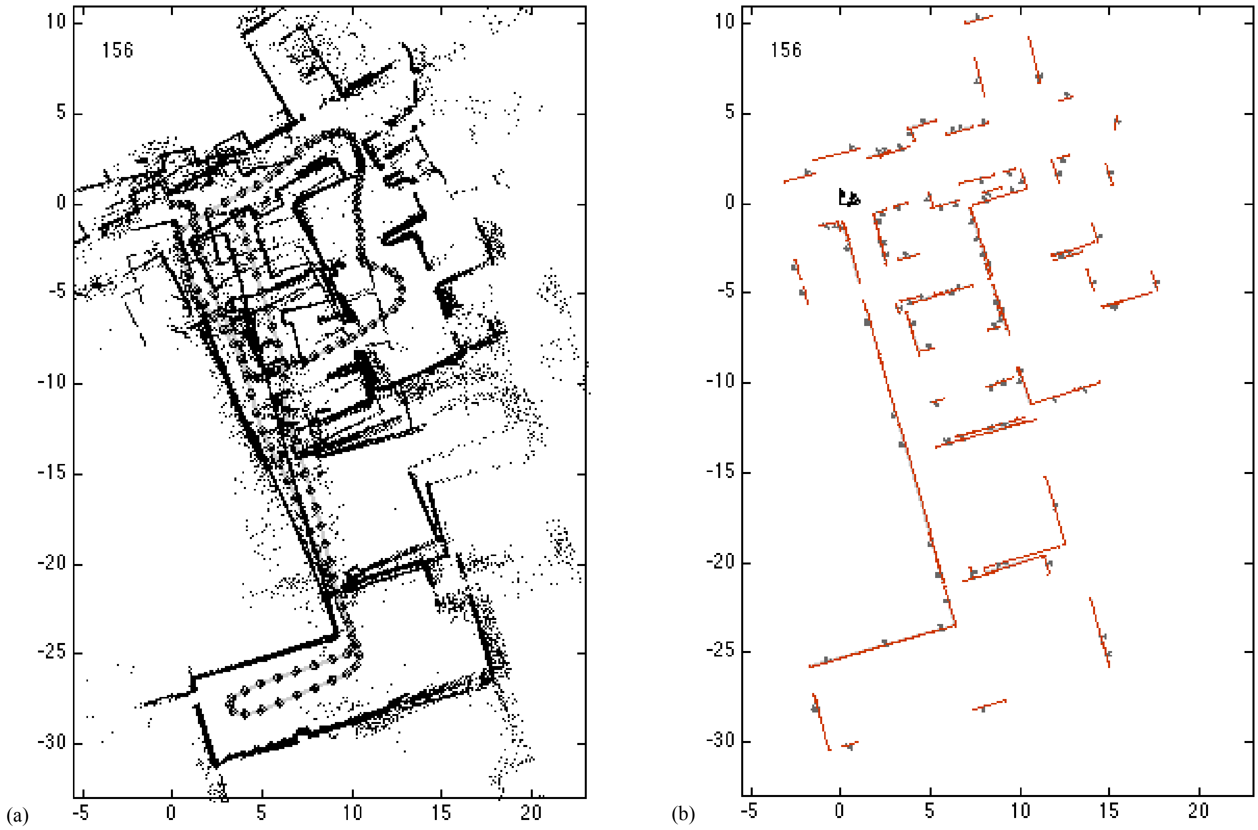

Figure 7. Mapping experiment 1: (a) virtual 2D scan and odometry, (b) line-based map calculated by SLAM algorithm

platform. The second difficulty is the lower data rate of virtual 2D scans in comparison to standard 2D sensors. In our experiments the observations are made about every 0.5m.

In spite of that the SLAM algorithm was able to calculate consistent maps for both test runs. This robustness can be attributed to the use of virtual 2D scans. The 360° opening angle of the scanner allows the complete observation of the surrounding. This is especially useful while passing doors, as features in the old room and in the new room can be seen in the same scan. Another essential fact benefit is the ability to observe also partially occluded walls. This contributes to the localization and mapping process as it provides landmarks that cannot be seen with a 2D sensor. The fact that obstacles and clutter are mostly removed from virtual 2D scans allows having more confidence in the sensed observation. For this reason the SLAM algorithm needs less observations to add a new feature into the map correctly.

We further observe an over-segmententation of the map. Walls are modeled by several, sometimes overlapping segments where one long segment could be expected. This type of inconsistency in the description of the environment is typical for single-segment features and has been observed in [10] and [12]. As this problem is already known from pure 2D system it cannot be led back to the new 3D perception.

V. Conclusion

This paper presented a novel method for robust localization and mapping of cluttered indoor environments. The new ability to model cluttered environments with line-feature maps is gained by the use of 3D perception. We introduced an algorithm to extract virtual 2D scans from full 3D laser range data. These virtual 2D scans that contain mainly static wall points are used as input data for a 2D line-based SLAM algorithm. By this means we presented an effective combination of rich 3D perception and efficient 2D localization and mapping. The applicability of this new method was demonstrated by experiments within industrial indoor environments.

Acknowledgment

The work described is part of a collaboration within the project MoRob (Modular Educational Robotic Toolbox), which is funded by the Swedish Wallenberg Foundation, the German Federal Ministry of Education and Research and the Research Ministry of Lower Saxony. Furthermore the authors would like to thank John Folkesson, Andreas Hedstrom, Patric Jensfelt, and Carl Lundberg for their help with the bakery experiments.

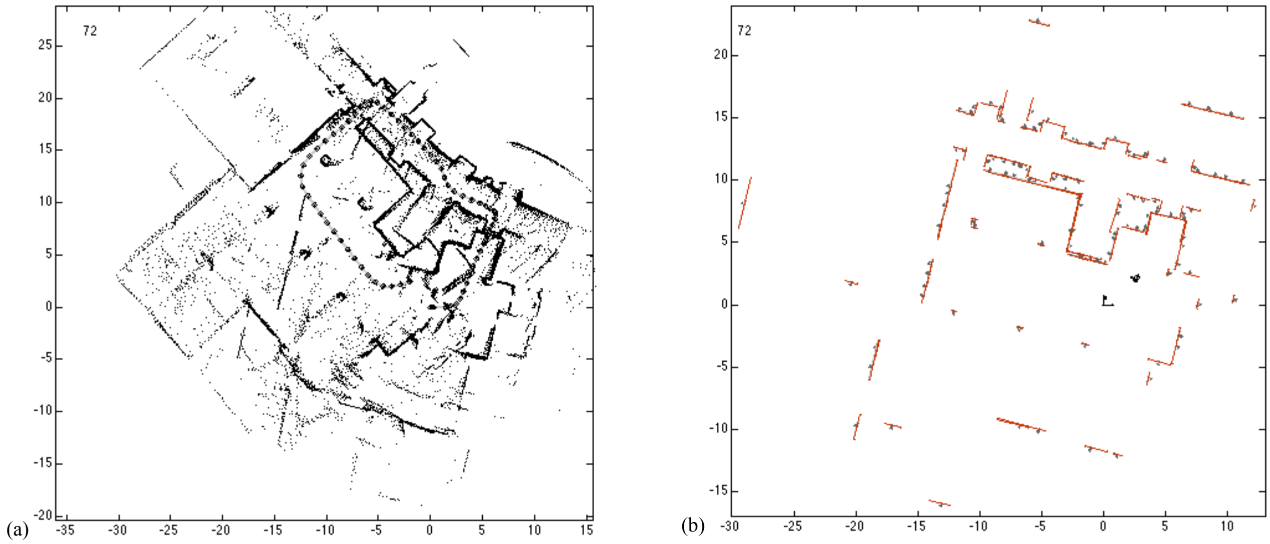

Figure 8. Mapping experiment 2: (a) virtual 2D scan and odometry, (b) line-based map calculated by SLAM algorithm

References

[1] J. Weingarten, G. Grüner, and R. Siegwart, "A Fast and Robust 3D Feature Extraction Algorithm for Structured Environment Reconstruction", International Conference on Advanced Robotics, Coimbra, Portugal, July 2003.

[2] H. Surmann, K. Lingemann, A. Nüchter, and J. Hertzberg, "A 3D Laser Range Finder for Autonomous Mobile Robots", International Symposium on Robotics, Zurich, Switzerland, April 2001.

[3] D. Hähnel, and W. Burgard, "Map Building with Mobile Robots in Populated Environments", IEEE/RSJ Conference on Intelligent Robots and Systems, Lausanne, Switzerland, September 2002.

[4] S. Lacroix, A. Mallet, D. Bonnafous, G. Bauzil; S. Fleury, M. Herrb, and R. Chatila, "Autonomous Rover Navigation on Unknown Terrains: Functions and Integration", International Journal of Robotics Research, vol. 21, issue 10, October 2002.

[5] P. Bellutta, R. Manduchi, L. Matthies, and K. Owens, "Terrain Perception for Demo III", IEEE Intelligent Vehicle Symposium, Dearborn, USA, October 2000.

[6] C. Brenneke, O. Wulf, and B. Wagner, "Using 3D Laser Range Data for SLAM in Outdoor Environments", IEEE/RSJ Conference on Intelligent Robots and Systems, Las Vegas, USA, October 2003.

[7] O. Wulf, and B. Wagner, "Fast 3D Scanning Methods for Laser Measurement Systems", International Conference on Control Systems and Computer Science, Bucharest, Romania, July 2003.

[8] P. Moutarlier, and R. Chatila, "Stochastic Multisensory Data Fusion For Mobile Robot Location and Environmental Modelling", International Symposium on Robotics Research, 1990.

[9] R. Smith, M. Self, and P. Cheeseman, "A Stochastic Map for Uncertain Spatial Relationships", 4th International Symposium on Robotics Research, 1987.

[10] J.A. Castellanos, and J.D. Tardos, "Mobile Robot Localization and Map Building: A Multisensor Fusion Approach", Kluwer, 1999.

[11] K.S. Chong, and L. Kleeman, "Accurate Odometry and Error Modelling for a Mobile Robot", IEEE International Conference on Robotics and Automation, Albuquerque, USA, 1997.

[12] K.O. Arras, "Feature-Based Robot Navigation in Known and Unknown Environment", PhD Dissertation N° 2765, Swiss Federal Institute of Technology Lausanne, Autonomous Systems Lab, June 2003.

[13] P. Hohmann, U.Gerecke, and B. Wagner, "A Scalable Processing Box for Systems Engineering Teaching with Robotics", International Conference on Systems Engineering, Coventry, UK, 2003.